DOCUMENT RESUME

ED 180 445 IR 007 919

AUTHOR Jones, Ernest L.
TITLE SNAP/SHOT Your Ability to Support That Next Application.
PUB DATE 6 Apr 79
NOTE 27p.: Paper presented at the Annual Conference on Academic Computing Applications (Gary, IN, April 6, 1979); For a related document, see IR 007 915
AVAILABLE FROM CAUSE, 137 29th Street, Boulder, CO 80303 (For reproduction costs)

EDRS PRICE MF01/PC02 Plus Postage.
DESCRIPTORS Computer Oriented Programs; *Data Analysis; *Data Collection; *Electronic Data Processing; Models; Networks; *Simulation

ABSTRACT

SNAP/SHOT (System Network Analysis Program-Simulated Host Overview Technique) is a discrete simulation of a network and/or host model available through IBM at the Raleigh System Center. The simulator provides an analysis of a total IBM Communications System. Input data must be obtained from RMF, SMF, and the CICS Analyzer to determine the existing environment and from analysis and users on anticipated applications and their volumes. The simulator permits the introduction of numerous variables in order to evaluate the impact of altering resources such as memory, software, paging devices, CPU's, lines and terminals, and altering volumes of transactions, message mixes, and size of transactions. Indiana University has used SNAP/SHOT twice to identify the volume of transactions that was supportable on existing equipment and the changes that would improve performance. After the first run, a group was assigned to optimize some transactions and additional memory was added to reduce the paging rate. The second run was made to see what had been accomplished and to simulate simultaneous student registration at the six small campuses, and then at the large campuses. Graphs and charts identify the many variables utilized in the processing of Indiana University data and show the results obtained. (Author/JEG)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED180445

U.S. DEPARTMENT OF HEALTH,
EDUCATION & WELFARE
NATIONAL INSTITUTE OF
EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

SNAP/SHOT YOUR ABILITY TO SUPPORT THAT NEXT APPLICATION

by

Ernest L. Jones

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Ernest L. Jones

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

IR007919

SNAP/SHOT YOUR ABILITY TO SUPPORT THAT NEXT APPLICATION

Ernest L. Jones, Director
Data Systems and Services
Indiana University

SNAP/SHOT (System Network Analysis Program-Simulated Host Overview Technique) is a discreet simulation of a network and/or host model available through IBM at the Raleigh System Center. The simulator provides an analysis of a total IBM Communications System including a maximum of 100 CPU's and their associated 370X, remote 370X, lines and terminals.

Input data must be obtained from RMF, SMF and the CICS Analyzer to determine the existing environment and from analysis and users on anticipated applications and their volumes. The simulator permits the introduction of numerous variables in order to evaluate the impact of altering resources such as memory, software, paging devices, CPU's, lines and terminals and altering volumes of transactions, message mixes, size of transactions, etc. The results provide response times by transaction, by line; shows CPU and line utilization, software and 3705 time. Bottlenecks are easy to spot, and the impact of changes to eliminate the bottlenecks can be evaluated by changing parameter and getting another run.

Indiana University has used SNAP/SHOT twice to identify what volume of transactions was supportable on existing equipment and what changes would improve performance. After the first run, a group was assigned to optimize some transactions; and additional memory was added to reduce the paging rate. The second run was made to see what was accomplished and to simulate simultaneous student registration at the six small campuses and then finally adding the large campuses.

Graphs and charts contained in the paper identify the many variables utilized in the processing of Indiana University data and show the results obtained.

SNAP/SHOT YOUR NEXT APPLICATION

If you have an uneasy feeling about the impact of some new, large project about to be placed on your computer--whether it is a new on-line transaction, RJE, lines and terminals, etc.--you might want to consider "SNAP/SHOTing" it. SNAP/SHOT (Systems Network Analysis Program-Simulated Host Overview Technique) is a discreet simulation of an IBM Communications network and/or host model. The simulator provides an analysis of a maximum of 100 CPU's and their associated 370X, remote 370X, paging devices, lines, terminals, communications protocol, and associated software.

To use SNAP/SHOT, the user provides a description of each application to be included in simple, macro terms such as the number of instructions executed, the number of DB/DC calls, the number of I/O's, line configurations, types and numbers of terminals, line speeds, communication controllers, modems, and paging devices. More details and sources of information will be discussed later. The first step is to establish a base level in the simulator results that compares favorably with observations of actual performance; therefore, you specify the central processor in use and the paging rate you are observing on your system at some selected time period. One item that is not specified, which seems strange at first, is memory size. After some thought, it is clear that paging rate is directly related to the amount of memory available and the philosophy of the installation on the use of memory.

After all input data has been provided and SNAP/SHOT has been executed, the host analysis section provides information on CPU utilization based on path lengths, interrupts, priorities, and the CPU type specified. The network section using the Emulator and/or Network Control Program line scheduling algorithm calculates line utilizations, message response times, and queues. This model can generate a diagram of the network described and provide the rate of messages generated for each location.

Overall message rates for the system are user specified as a series of parameters prior to each run, thus allowing future or peak loads to be simulated.

Indiana University has utilized SNAP/SHOT twice. The first time was in March, 1977, and the second time in September, 1977. We had discussed an earlier use, but could not supply information in sufficient detail to drive the model.

The sequence of events that led to the first use began when several on-line systems were implemented in quick succession in the summer of 1976. Each of these systems when tested on the test data bases with test volumes seemed to operate within acceptable parameters, but when thrown together with production volumes was completely unacceptable. Tuning efforts were continuous during the fall and winter months, but response time on the student records maintenance transactions remained at 18 seconds average with wide fluctuations.

Some equipment and software changes had been identified which we thought would improve response time, but we could only speculate on the degree of improvement that would result. We had identified that the paging rate during the time the on-line systems were operational was higher than it should be. A crash program had been proposed to review all major transactions for the purpose of optimizing their performance, but again we were not sure how much we would have to improve these programs to make significant improvement. A goal of eight second response time was set on the complex Student Records Maintenance and Admissions transactions. Retrievals were to be in the two to four second response-time range.

Armed with our input for the model, our IBM Systems Engineer headed for Raleigh. A number of sources had been tapped to accumulate the data, including software packages, system programmers, application analysts and users. As mentioned above, a crash program had been started to identify response time problems, so a large part of the technical fact gathering was in place or completed.

Since users are an important part of any computer application, let's look at their participation in the projection of the work loads.

The registrar provided a list of each regional campus, the number of students to be enrolled, the number of days and hours per day of registration. With these figures the number of students to be processed per hour and the required number of terminals were calculated.

The application analyst worked with the programmers to determine the number of CICS messages and tasks to perform a complete student registration. Combined with the above statistics a current message rate, as well as a growth factor, resulted in a message rate for each campus by line and terminal. The 3705 Emulation Program (EP) Generation provided each existing line, speed, number and types of terminals, and type of modem used.

The SMF listings for each RJE station recorded the number of cards read, lines (characters) printed and jobs run by given time periods. Several days were processed and the peak periods were used for the simulation. Also the batch requirements were obtained from the SMF output.

CICS Performance Analyzer (PA) was used for many needed facts about the transactions and terminals. A report by terminal gave us the current transaction or application mix and the number of characters in and out. This concluded the data necessary to define our network to the simulator.

PA provided most of the needed information about the applications. A listing by application gave us the CPU seconds, from which we could calculate the approximate number of instructions executed, the number of I/O operations and the number of tasks completed. The total number of tasks was used to approximate the total CICS load by application.

Together with the system programmers we completed the CPU and Systems definitions.

From all of the above sources the needed input for the simulator was then encoded in the proper macros.

Before we left for Raleigh a meeting with system programming, application systems and the SE refined the data to our best estimate for both the current system and the projected changes.

The primary goal of the March visit to Raleigh was to pinpoint what was causing our response time problems and to identify where our efforts should be concentrated to help alleviate the problems.

Once the data was input and we understood the procedures necessary to execute the program, we set out to firmly establish the base run that would represent--as closely as possible--our observation of the actual system.

Fortunately the Raleigh Center provided a consultant, who was one of the authors of the system, to answer questions as they arose. On Monday and Tuesday we called him frequently on technical as well as philosophical questions. By Wednesday morning we felt we had established a base level. We later confirmed this by calling the Assistant Director of System Programming and Operations and running a controlled experiment for a four-hour period where we eliminated a large portion of our batch initiators and attempted to control our paging rate at 15 pages a second.

Prior to describing some of the capabilities of SNAP/SHOT that we used, it is necessary to describe the form that was designed to record the data for comparison purposes. Please see the example and explanation in Appendix A.

Forty runs were made in three days and they are described in Appendix B. A review of that list of runs will indicate that we compared hardware, operating systems, memory size (paging rate), line speeds and protocol, and paging devices.

The effect of holding all other factors constant while reducing the paging rate (increasing available memory) is shown in Graph 1. This action alone will not "save the ship" but will give the crew some time to patch the holes. This is exactly what was done. A software team was established to review all transactions and make changes to improve their performance. The number of keyed elements on the data base was reduced from 69 to 22; and every program was run through OPTIMIZER II, a product of CAPEX. The team, affectionately known as the Software Attack Team or SWAT, accomplished more than had been hoped. This will be described later.

Graphs two, three, and four represent simulations of different central processors at 10 and 20 pages a second. Obviously the Attached Processor provided considerable improvement in performance over the Unit Processor in both on-line response time and batch throughput, and at ten pages a second outperformed the 158MP.

The 168-1 and 168-3 obviously would be the machines to have to improve response time without requiring a lot of programmer time. However, funds were not available; and we were not on IBM's delivery schedule.

Although some runs were made, little effort was spent reviewing line speeds and organizations because it was obvious we were unable to sustain a message rate high enough to tax the existing line configuration. One area of the output of SNAP/SHOT identifies the point of major delay as shown in Chart 1. If you read from right to left beginning with the column headed "APPL PROG," you can see the time delay at each point. This particular example identifies lines as the major problem.

The second trip, made in September, 1977, had two goals: 1) Evaluate the improvements made through SWAT and hardware additions (1 meg. of memory) and 2) to determine if the on-line registration system could be supported on all campuses when the AP was installed. The feeling was that the largest campus with 32,000 students was out of the question. The Indianapolis campus with 26,000 students was doubtful, but the small campuses could be handled.

Since we had become "experts" during our first trip to Raleigh, we were scheduled for Monday-Wednesday on the second trip. We had ample time to accomplish our goals. Twenty-one runs were completed which are described in Appendix C.

The first seven runs were made on various mainframes after establishing the base run on the 158-3. Hardware simulated was 158-3AP, 168-1, 168-3, 168-3AP, and 3033. The operating system we used in all cases was MVS, and the paging rate was held to 10 pages per second. These decisions were firm after our first trip. Increased message rates were used to reflect a growing environment.

One of our runs during our first trip used a 15% reduction in the number of instructions executed in each transaction which indicated that response time could be improved considerably if the transactions were optimized, and the Software Attack Team set about to do just that.

Graph 5 represents simulations before and after SWAT. We held all factors constant except the description of the program logic which we changed based on the improved versions of the programs as a result of our optimizing efforts. Work was done on both on-line transactions and batch programs. The on-line results represent the improvement made in the Student Records Maintenance transactions. Again the results were confirmed through observations made in actual operational situations.

Working from the new base, the changes in central processors produced the results in Graph 6. The 158AP deteriorates very quickly as the message rate increases. Both on-line and batch capability are impacted. The 3033 maintains the best response time of all systems, but the 168AP provides additional batch capability. From a cost/

performance ratio, the 3033 is still the best buy. If we see some 168AP machines appear in the third-party market, the cost/performance gap may close.

At this point we added the Registration transactions and increased the message rate to include what we anticipated the volume would be if all six small, remote campuses were to register simultaneously. These campuses range from 700 to 6300 students. We limited our simulations to the 158AP which we planned to install in December, 1977, and the 3033 which was the machine size chosen to take us into the future.

We estimated that we would experience a message rate of 1.0 - 1.1 per second, and the results indicated that the 158AP should handle this message rate without significant degradation. The 3033 would show no impact on response time at that level.

The Indianapolis campus was then added to the transaction mix which we felt would generate 1.3 - 1.5 messages per second. It is obvious from Graph 7 that the 158AP would produce undesirable results. If the message rate did reach 1.5 per second, erratic conditions would exist with a 45 second response time and a standard deviation of 8.59. The 3033, on the other hand, remains relatively constant.

Adding the Bloomington campus to the "registration" mix produced some interesting results. At a message rate of 1.7, the response time was almost 8 seconds greater than at 1.7 messages per second with the small campuses plus Indianapolis. Because the registration transactions are larger in terms of resources required than the transactions normally used at registration by the Bloomington campus, communication lines became the constraining factor. A quick look at factors presented by SNAP/SHOT led us to experiment with changing from BTAM to NCP/VTAM and evaluating the effects of the new 32XX terminals using SDLC.

The ability to multidrop cathode ray tubes and RJE stations led to a line configuration which indicated we could go from 22 lines to six 9600 baud lines and drop our response time on the full registration implementation to 2.70 seconds at 1.35 messages per second.

Where are we now? This past semester with two small campuses on the registration system, we experienced erratic response times on the 158AP. What happened? Prior to September, 1977, our average message rate per second for a period of a month was .46 to .50--obviously with peaks and valleys--so we forecast a doubling of that rate for registration. An analysis of the three days of registration this fall uncovered a significantly higher message rate, reaching 2.18 messages per second in one 15-minute period but averaging during the three days better than 1.5 per second. New transactions have been added, and the Bloomington campus' use of on-line transactions during registration was significantly higher than anticipated.

We are hoping to get another SNAP/SHOT this winter in order to apply our new information on transaction volumes and try to forecast

what is going to happen when we add a new Purchasing Inquiry system, a Student Loan system, Financial Aids system, Course Maintenance, etc.

The synergistic effect could be devastating. Transactions tested on Test CICS and approved, can, when put with all other transactions on Production CICS, cause serious problems. We plan to push the 3033, lines, paging devices, etc. to their limit to see what growth potential we have with this machine, which will be installed in July, 1979.

In summary, SNAP/SHOT provides an opportunity to look into the future at what impact new applications or growth of existing ones will have on your ability to service your users and where you might look to make changes in hardware or software that would allow you to cope with the new environment.

NOTES ON READING OUTPUT REPORTS

The following is a description of the comparison charts which are used in the report:

Columns	Explanation
1	Run number - each simulation was given a unique number which can then be used as a quick reference. The run numbers and description of each follows the charts.
2	Paging Rate - the paging rate of the CPU for a given simulation.
3	MSGS/SEC - the actual number of messages per second for the given simulation.
4	RESPONSE TIME - RM - the average total response time for an "RM" transaction. Total time is the sum of application time, 3705 time and line time for "RM" transaction only.
5	Std Dev - the value of one standard deviation from the average total response time of an "RM" transaction.
6	AD - the average total response time for and "AD" transaction.
7	Std Dev - the value of one standard deviation from the average total response time of an "AD" transaction.
8	CPU Utiliz - the average CPU utilization for the duration for this particular simulation. (In the case of an Attached Processor System there are two CPU's; and thus, two utilization percentages.)
9	BATCH COMPL - based on the criteria of the advanced job, this column is total of batch jobs completed in all batch regions during the simulation.
10	CICS & DB Util - the sum of utilization for the CICS/application region and the data base region.
11	RM APL TIME - the time it took the "RM" application to execute in the CPU.

Columns	Explanation
12	AD APL TIME - the time it took the "AD" application to execute in the CPU.
13	LINE 34 RM - the total response time of an "RM" transaction on line 34 only.
14	LINE 34 AD - the total response time of an "AD" transaction on line 34 only.

PERFORMANCE OF 158-3 UP SVS WITH VARYING RATES

Run Number	Paging Rate	Msgs./ Sec.	Response Time				CPU Utiliz.	Batch Compl.	CICS & DB Util	RM Apl. Time	AD Apl. Time	Line 34 RM AD
			RM	StdDev	AD	StdDev						
1	20	.41	12.9	3.4	15.0	5.0	93.4	50.5				
		.46	18.0	11.0	26.0	14.6	94.4	44.7				
		.51	24.0	14.7	28.7	16.4	95.6	37.2				
		.67	57.9	22.3	58.7	22.8	97.9	20.2				
38	15	.41	11.5	2.9	13.8	5.1	93.0	51.7				
		.46	14.2	8.5	21.9	11.9	94.2	47.7				
		.51	18.6	10.1	22.9	10.1	95.1	38.6				
		.67	42.3	21.1	43.4	21.7	98.2	17.2				
4	10	.41	10.5	2.4	12.5	4.7	92.8	52.4				
		.46	12.6	7.7	17.9	9.1	93.5	50.2				
		.51	17.2	9.7	20.2	9.1	94.6	40.0				
		.67	34.4	18.2	36.7	18.6	98.5	13.9				

RUN NO.	CPU TYPE	OPERATING SYSTEM	PAGING RATE	DESCRIPTION
1	158-3 UP	SVS	20	Base (=message mix and terminal configuration of existing environment).
	Purpose:	Establish performance criteria for current system.		
2	158-3 UP	MVS	20	Base.
	Purpose:	Determine impact on current system of changing to MVS operating system without adding additional memory.		
3	158-3/AP	MVS	20	Base.
	Purpose:	Establish performance criteria for current loads on an AP system with a paging rate of 20.		
4	158-3 UP	SVS	10	Base.
	Purpose:	Same as #1 with reduced paging rate (indicative of additional memory).		
5	158-3 UP	MVS	10	Base.
	Purpose:	Same as #2 with lower paging rate.		
6	158-3/AP	MVS	10	Base.
	Purpose:	Same as #3 with lower paging rate.		
7	158-3/AP	MVS	10	Increased message rates on base.
	Purpose:	Determine performance of AP at higher utilization.		
8	Not Used	(See Note 1)		
9	Not Used	(See Note 1)		
10	158-3 UP	SVS	10	Addition of VTAM/NCP and SDLC on base.
	Purpose:	To determine impact of an SNA environment on current system.		
11	158-3 UP	SVS	10	Elimination of all batch work.
	Purpose:	Evaluate impact of changing current system to all TP environment.		

RUN NO.	CPU TYPE	OPERATING SYSTEM	PAGING RATE	DESCRIPTION
12	158-3/AP	MVS	10	Increased message rates on base.
	Purpose:	Add additional points for graphing purposes to Runs 6 and 7.		
13	158-3/AP	MVS	20	Use drum for a paging device.
	Purpose:	Determine significance of a drum in a high paging environment.		
14	168-3	MVS	20	Base.
	Purpose:	Determine performance of workload on a 370/168 Model 3.		
15	168-1	MVS	20	Base.
	Purpose:	Same as #14 with a 370/168 Model 1.		
16	158-3/AP	MVS	20	Base.
	Purpose:	Evaluate performance of an AP with no batch running (see #11).		
17	158-3/AP	MVS	20	Increased message rates.
	Purpose:	Same as #7 with a higher paging rate.		
18	158-3/MP	MVS	10	Base.
	Purpose:	Evaluate MP performance.		
19	158-3/AP	MVS	10	Change to a DL1/CICS environment with HDAM files.
	Purpose:	Experiment with a DL1 interface with CICS using HDAM file construction in the data base.		
20	158-3/AP	MVS	20	Base with 15% fewer instructions.
	Purpose:	Simulate effect of a 15% greater efficiency in application programs.		
21	158-3AP	MVS	30	Base with 15% fewer instructions.
	Purpose:	Same as #20 with higher paging rate.		

RUN NO.	CPU TYPE	OPERATING SYSTEM	PAGING RATE	DESCRIPTION
22	158-3 UP	SVS	20	No batch.
	Purpose: Same as #11 with higher paging rate.			
23	158-3 UP	SVS	10	Only 3 batch initiators.
	Purpose: Determine effect on current system of cutting back to only 3 initiators at a low paging rate.			
24	Not used (see note 2)			
25	168-1	SVS	20	Base.
	Purpose: Same as #15 with SVS operating system.			
26	158-3/AP	MVS	10	Increased message rates on base.
	Purpose: Same as #7			
27	158-3/AP	MVS	10	Maintaining current message rates, line 34 was changed to 1 control unit with 24 terminals running at 4800 baud, SDLC.
	Purpose: Determine impact of SDLC in environment described.			
28	158-3/AP	MVS	10	Same as #27 at 9600 baud, SDLC.
	Purpose: Determine impact of higher baud rate with SDLC.			
29	158-3/AP	MVS	10	Same as #27 at 4800 baud, bisync.
	Purpose: Determine performance of environment described in Run #27 with existing 4800 baud, bisync facilities.			
30	158-3 UP	SVS	10	Change to a DL1/CICS environment with HDAM files.
	Purpose: Experiment with a DL1 interface with CICS using HDAM file construction in the data base, running on current equipment.			
31	158-3 UP	SVS	10	Same as #30, but using HIDAM files.
	Purpose: Same as #30, but with HIDAM file construction in data base.			
32	158-3/AP	MVS	10	Same as #27 at 9600 baud, bisync.
	Purpose: Determine impact of increasing to 9600 baud, bisync in environment described in #27.			

RUN NO.	CPU TYPE	OPERATING SYTEM	PAGING RATE	DESCRIPTION
33	158-3/AP	MVS	20	Using a 3350 for a paging device.
	Purpose:	Determine significance of using a 3350 in a high paging environment.		
34	158-3AP	MVS	10	New mix of transactions with additional terminals added (total of 158 terminals).
	Purpose:	Determine performance of AP in future anticipated environment.		
35	158-3/AP	MVS	10	Message rates of approx. 50% increased volume on line 34 (1 control unit, 26 terminals) run at 4800 baud, bisync.
	Purpose:	Determine performance of a 4800 baud, bisync line in environment described.		
36	158-3/AP	MVS	10	Change to a full IMS DB/DC environment.
	Purpose:	Experiment with AP performance in a full IMS environment.		
37	158-3/AP	MVS	10	Same as #35 with a 9600 baud, bisync line.
	Purpose:	Same as #35 at 9600 baud, bisync.		
38	158-3 UP	SVS	15	Base.
	Purpose:	Attempt to duplicate exact situation being experienced on current system on 3/17/77.		
39	158-3/AP	MVS	10	New mix of transactions, additional terminals, local lines.
	Purpose:	Determine significance of putting some terminals on local lines.		
40	158-3/AP	MVS	10	Same as #35 with a 4800 baud, SDLC line.
	Purpose:	Same as #35 with a 4800 baud, SDLC line.		

Note 1: Initial attempts to determine significance of 9600 baud lines, but not properly configured.

Note 2: An initial run to determine significance of a 3350, but run at an improper paging rate.

RUN NO.	CPU TYPE	MESSAGE MIX	DESCRIPTION
1	158-3	C	Base (message mix, load, and terminal configuration of existing environment).
	Purpose:	Establish performance criteria of current system.	
2	158-3AP	C	Base.
	Purpose:	Establish performance criteria for current loads with the AP installed.	
3	158-3AP	C	Base with increased message rates of current mix.
	Purpose:	Determine if there is a breaking point on the AP with current mix.	
4	3033	C	Base.
	Purpose:	To provide a comparison of the 158 and the 3033, also to establish a base for the 3033.	
5	168-1	C	Base.
	Purpose:	To establish if the 168-1 should be in the migration plan.	
6	168-3	C	Base.
	Purpose:	To establish if the 168-3 should be in the migration plan.	
7	168-3AP	C	Base.
	Purpose:	To establish if the 168-3AP should be in the migration plan.	
8	158-3AP	CR	Base with registration mix for regional campuses.
	Purpose:	To determine the load on the CPU with regional campus registration and the respective line utilizations.	
9	3033	CR	Base with registration mix for regional campuses.
	Purpose:	Same as #8 on a 3033	

RUN NO.	CPU TYPE	MESSAGE MIX	DESCRIPTION
10	158-3AP	CRI	Base with registration mix for regional campuses and IUPUI.
	Purpose:	Same as #8 with addition of IUPUI.	
11	3033	CRI	Base with registration mix for regional campuses and IUPUI.
	Purpose:	Same as #9 with the addition of IUPUI.	
12	Not Used		
13	3033	T	Base with registration mix for all campuses including Bloomington.
	Purpose:	To determine the load on the CPU and lines for full registration.	
14	158-3AP	T	Base with full registration and NCP/VTAM, SDLC and new terminals.
	Purpose:	To determine impact of new terminals under SDLC on the 158-3AP.	
15	3033	T	Base with full registration mix, BTAM and old terminals.
	Purpose:	Same as #14 on a 3033.	
16	Not Used		
17	3033	T	Base with full registration mix, VTAM, new terminals, and increased batch.
	Purpose:	To determine if additional batch loads could be handled and the impact on TP.	
18	3033	T	Base with full registration and NCP/VTAM, SDLC and new terminals and new line configurations at 4800 baud.
	Purpose:	To establish a base of multidrop lines and their impact on response time.	

RUN NO.	CPU TYPE	MESSAGE MIX	DESCRIPTION
19	3033	T	Same as #18

Purpose: Same as #18 but with a simulation of 30 minutes.

20 Not Used

21	3033	T	Same as #18 with 9600 baud.

Purpose: To determine the benefits of increased line speed.

Message Mix Explanation:

C = Current message mix.

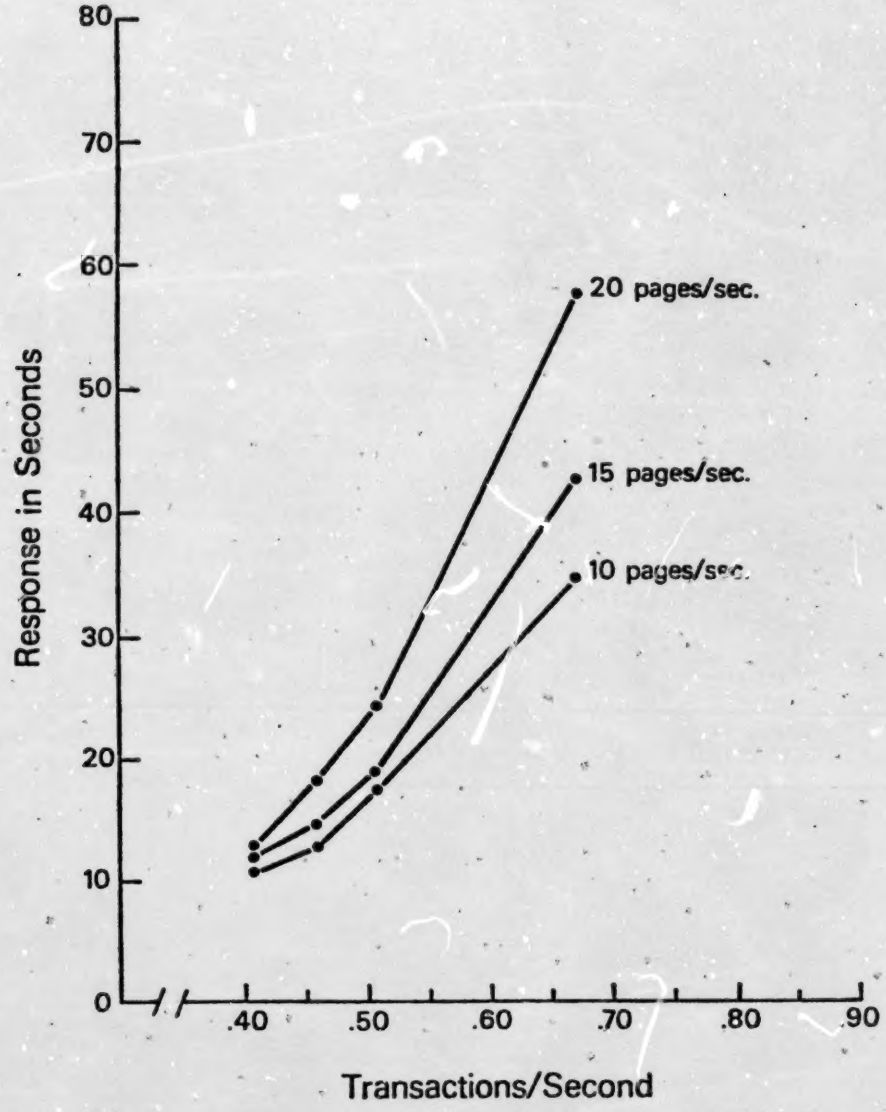
CR = Current message mix with predicated regional campus registration load.

CRI = Current message mix with predicted regional campus and IUPUI registration load.

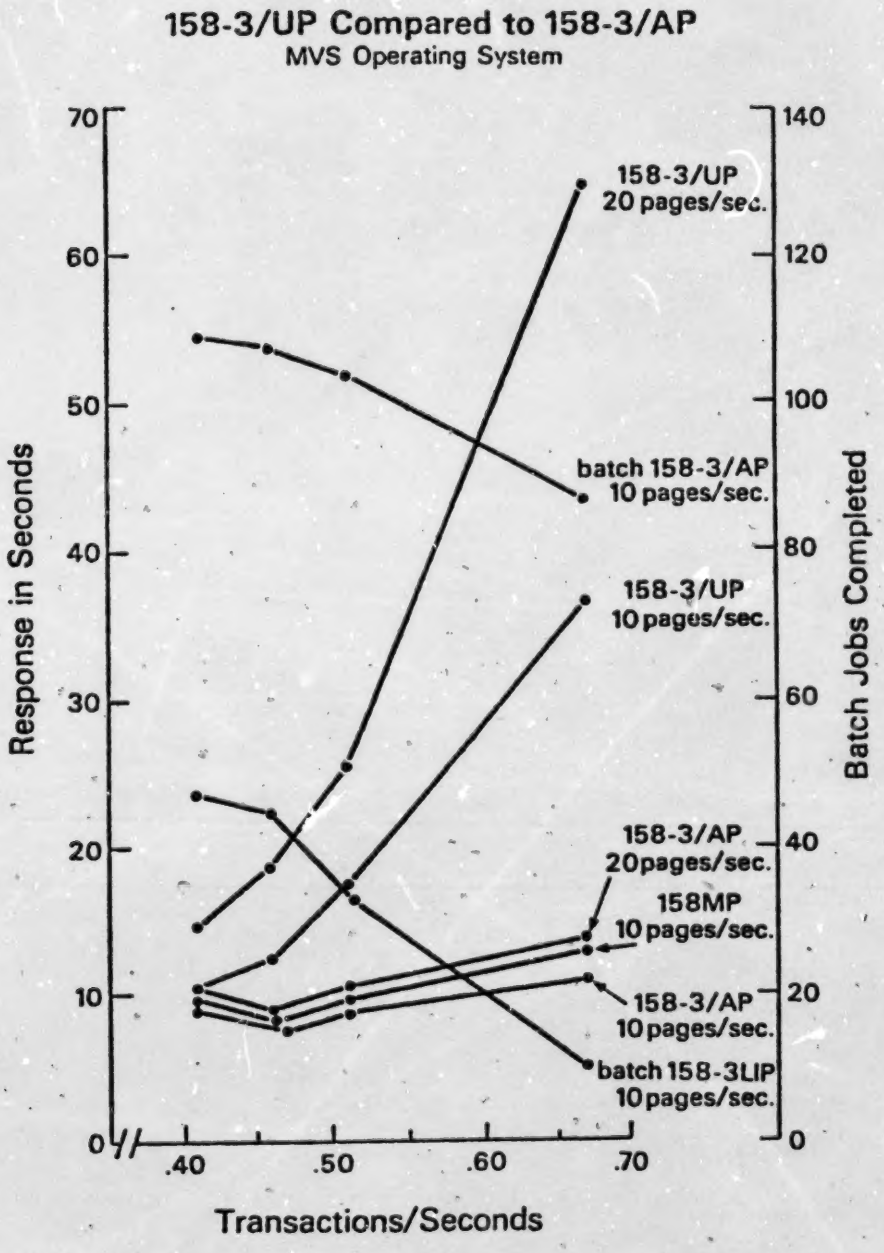
T = Current message mix with full predicted registration load at all campuses including Bloomington.

Effect of Lower Paging Rate (Increased Memory)
Records Maintenance Transaction

158-3 Unit Processor
SVS Operating System

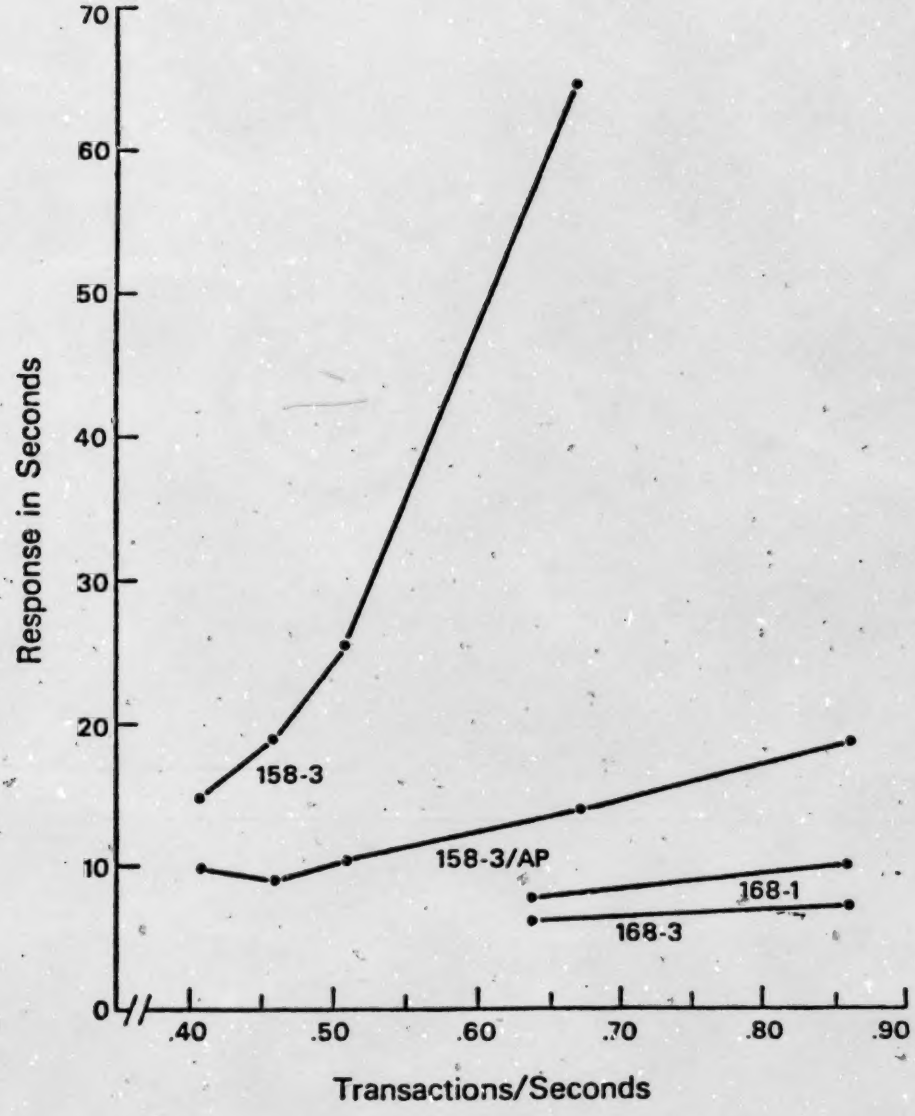


Graph 1

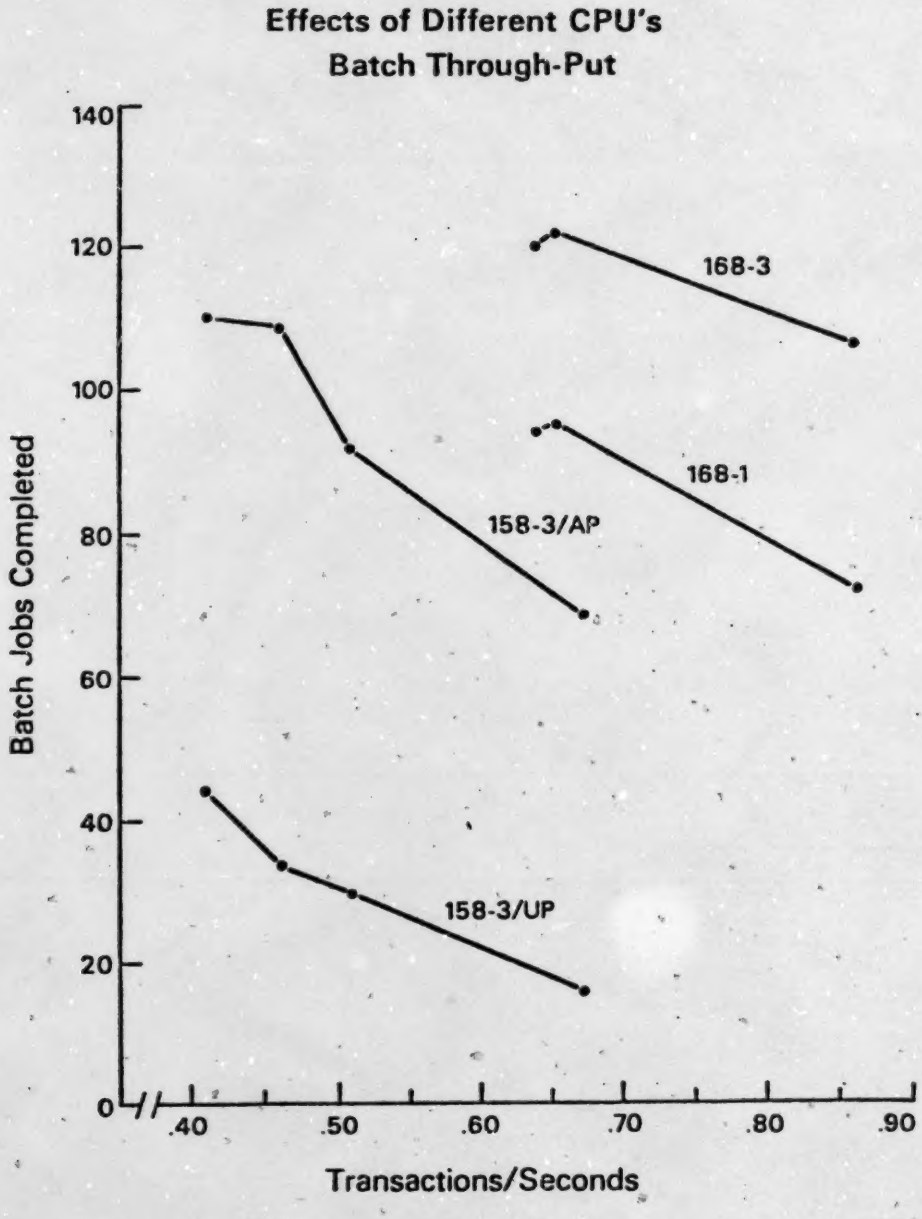

158-3/UP Compared to 158-3/AP
MVS Operating System
Response in Seconds
Batch Jobs Completed
Transactions/Seconds
70
60
50
40
30
20
10
0
140
120
100
80
60
40
20
0
.40
.50
.60
.70
158-3/UP
20 pages/sec.
batch 158-3/AP
10 pages/sec.
158-3/UP
10 pages/sec.
158-3/AP
20pages/sec.
158MP
10 pages/sec.
158-3/AP
10 pages/sec.
batch 158-3LIP
10 pages/sec.

Graph 2

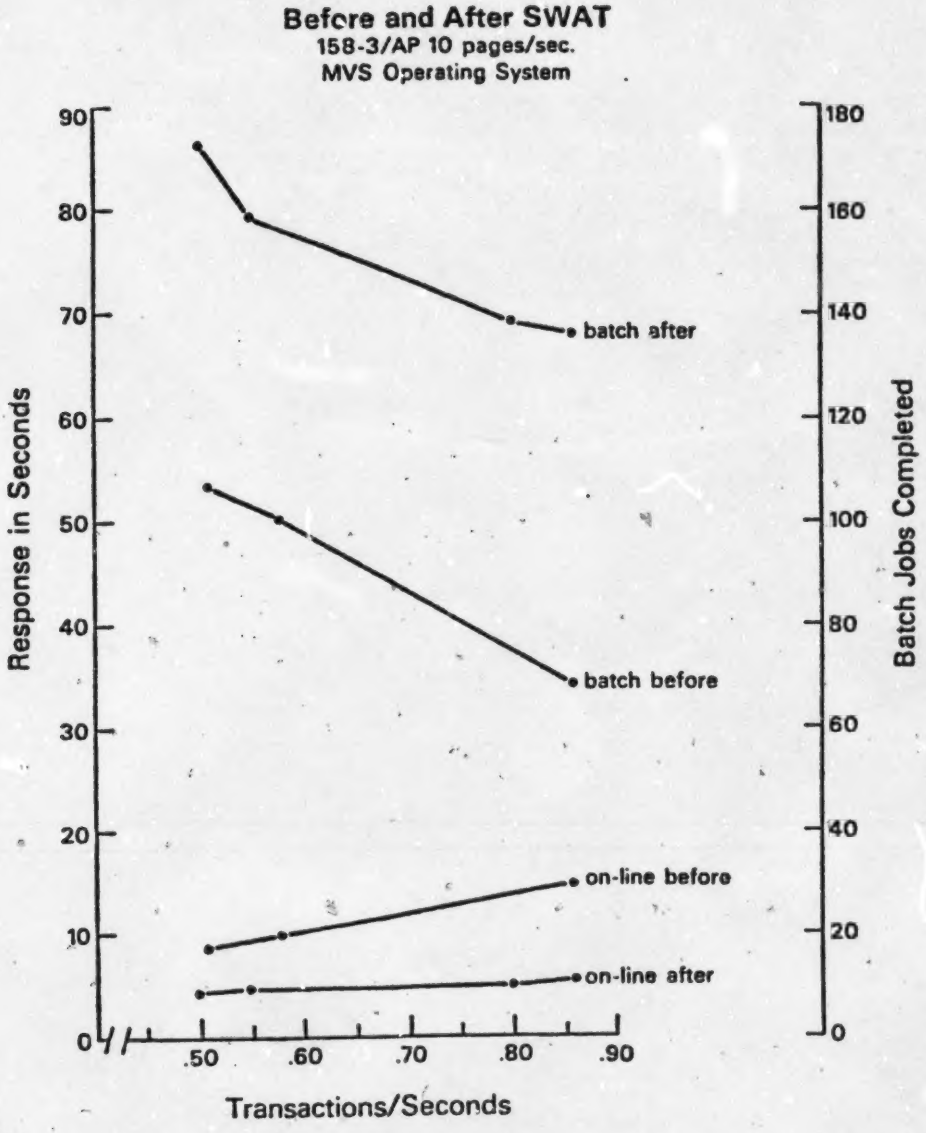
Effects of Different CPU's
Records Maintenance Transaction



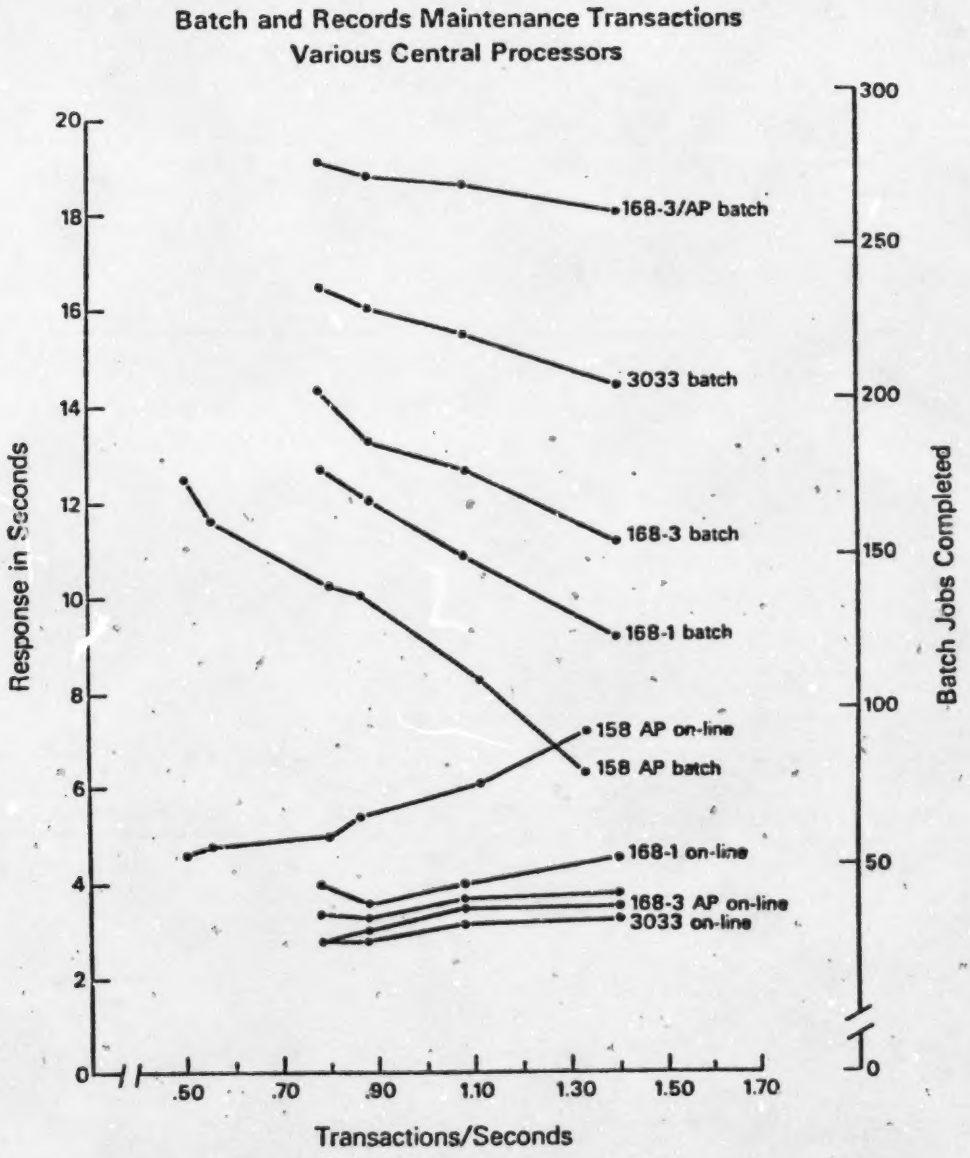
Graph 3


Effects of Different CPU's
Batch Through-Put
Batch Jobs Completed
140
120
100
80
60
40
20
0
168-3
168-1
158-3/AP
158-3/UP
.40
.50
.60
.70
.80
.90
Transactions/Seconds

Graph 4


Before and After SWAT
158-3/AP 10 pages/sec.
MVS Operating System
Response in Seconds
Batch Jobs Completed
Transactions/Seconds
batch after
batch before
on-line before
on-line after
90
80
70
60
50
40
30
20
10
0
180
160
140
120
100
80
60
40
20
0
.50
.60
.70
.80
.90

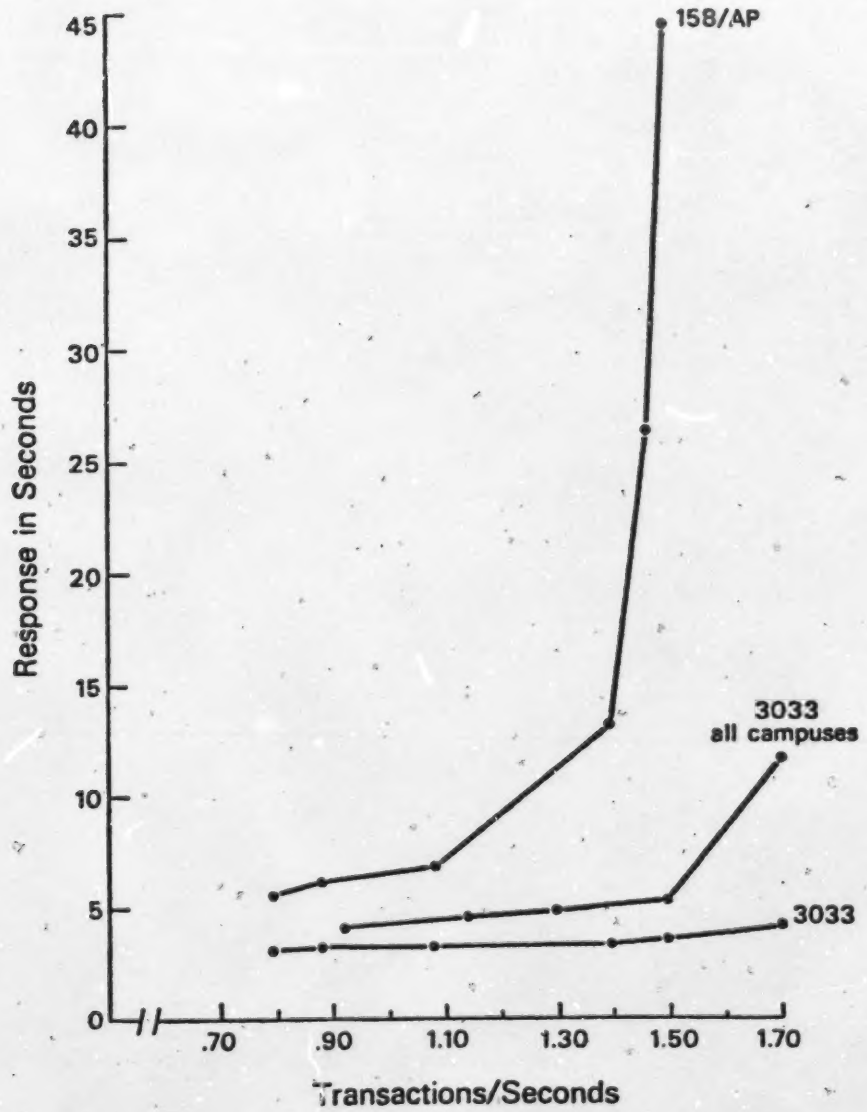
Graph 5


Batch and Records Maintenance Transactions
Various Central Processors
Response in Seconds
Batch Jobs Completed
Transactions/Seconds
20
18
16
14
12
10
8
6
4
2
0
300
250
200
150
100
50
0
.50
.70
.90
1.10
1.30
1.50
1.70
168-3/AP batch
3033 batch
168-3 batch
168-1 batch
158 AP on-line
158 AP batch
168-1 on-line
168-3 AP on-line
3033 on-line

Graph 6

Registration at the Small Campuses and Indianapolis

158/AP vs 3033 and 3033 All Campuses



Graph 7

RESPONSE TIMES FOR EACH NAMED MESSAGE IN SECONDS.

MESSAGE.NAME	TOTAL SAMPLE	LINE	3705	3705 NET-WORKING	CPU	CPU NET-WORKING	APPL PROG
ADMSG	45	7.012	4.069	3.820	3.818	3.814	3.814
CLMSG	29	5.245	3.870	3.504	3.503	3.499	3.499
RMMSG	33	5.774	3.962	3.709	3.709	3.706	3.706
SCMSG	3	6.143	4.177	3.774	3.774	3.770	3.770
SSMSG	17	6.683	3.919	3.854	3.854	3.851	3.851

Chart 1